Syntax of propositional logic

An atomic formula has the form A_i where $i = 1, 2, 3, \ldots$.
Formulas are defined by the following inductive process:

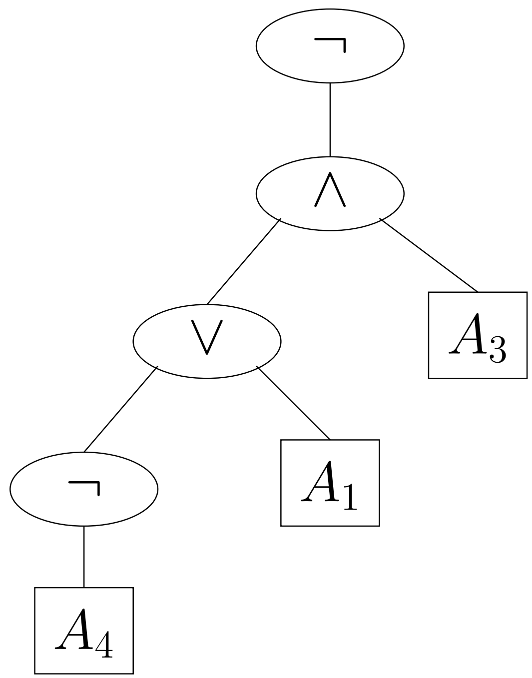
1. All atomic formulas are formulas
2. For every formula F, $\neg F$ is a formula.
3. For all formulas F und G, $(F \wedge G)$ and $(F \vee G)$ are formulas.

For $(F \wedge G)$	we say	F and G, conjunction of F and G
For $(F \vee G)$	we say	F or G, disjunction of F and G
For $\neg F$	we say	not F, negation of F

Syntax tree of a formula

Every formula can be represented by a syntax tree.

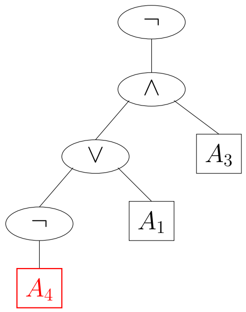
Example: $F = \neg((\neg A_4 \vee A_1) \wedge A_3)$



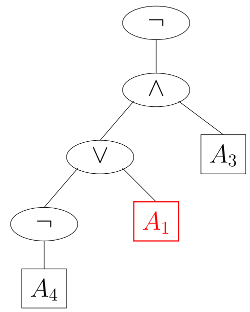
Subformulas

The subformulas of a formula are the formulas corresponding to the subtrees of its syntax tree.

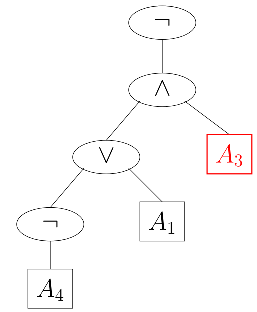
A_4



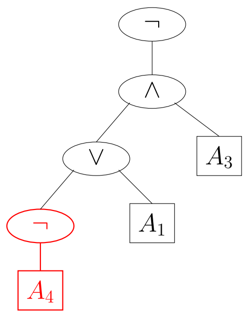
A_1



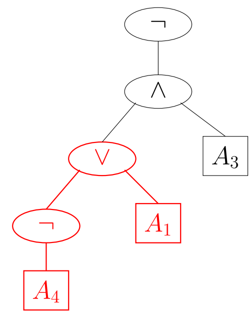
A_3



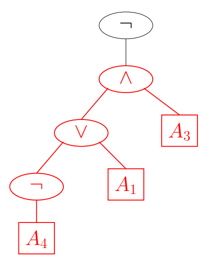
$\neg A_4$



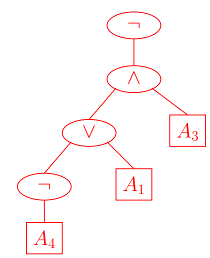
$(\neg A_4 \vee A_1)$



$((\neg A_4 \vee A_1) \wedge A_3)$



$\neg((\neg A_4 \vee A_1) \wedge A_3)$



Semantics of propositional logic (I)

The elements of the set $\{0, 1\}$ are called truth values.

An assignment is a function $\mathcal{A}\colon D \to \{0, 1\}$, where D is any subset of the atomic formulas.

In case $D = \{A_i \mid i \in X\}$ for some finite set $X \subset \mathbb{N}$ we sometimes also write $\{(A_i, \mathcal{A}(A_i)) \mid i \in X\}$ to denote $\mathcal{A}$.

We extend $\mathcal{A}$ to a function $\hat{\mathcal{A}}\colon E \to \{0, 1\}$, where $E \supseteq D$ is the set of formulas that can be built up using only the atomic formulas from D.

Semantics of propositional logic (II)

$$\hat{\mathcal{A}}(A) = \mathcal{A}(A) \quad \text{if } A \in D \text{ is an atomic formula}$$

$$\hat{\mathcal{A}}((F \wedge G)) = \begin{cases} 1 & \text{if } \hat{\mathcal{A}}(F) = 1 \text{ and } \hat{\mathcal{A}}(G) = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{\mathcal{A}}((F \vee G)) = \begin{cases} 1 & \text{if } \hat{\mathcal{A}}(F) = 1 \text{ or } \hat{\mathcal{A}}(G) = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{\mathcal{A}}(\neg F) = \begin{cases} 1 & \text{if } \hat{\mathcal{A}}(F) = 0 \\ 0 & \text{otherwise} \end{cases}$$

We write $\mathcal{A}$ instead of $\hat{\mathcal{A}}$.

Truth tables (I)

We can compute $\hat{\mathcal{A}}$ with the help of truth tables.

Observe: $\hat{\mathcal{A}}(F)$ depends only on the definition of $\mathcal{A}$ on the atomic formulas that occur in F.

Tables for the operators $\vee$, $\wedge$, $\neg$:

A	B	A	$\vee$	B
0	0	0	0	0
0	1	0	1	1
1	0	1	1	0
1	1	1	1	1

A	B	A	$\wedge$	B
0	0	0	0	0
0	1	0	0	1
1	0	1	0	0
1	1	1	1	1

A	$\neg$	A
0	1	0
1	0	1

Abbreviations

$A, B, C,$

$P, Q, R,$ or $\dots$ for $A_1, A_2, A_3 \dots$

$(F_1 \rightarrow F_2)$ for $(\neg F_1 \vee F_2)$

$(F_1 \leftrightarrow F_2)$ for $((F_1 \wedge F_2) \vee (\neg F_1 \wedge \neg F_2))$

$(\bigvee_{i=1}^{n} F_i)$ for $(\dots((F_1 \vee F_2) \vee F_3) \vee \dots \vee F_n)$

$(\bigwedge_{i=1}^{n} F_i)$ for $(\dots((F_1 \wedge F_2) \wedge F_3) \wedge \dots \wedge F_n)$

$\top$ or $\mathbf{true}$ or 1 for $(A_1 \vee \neg A_1)$

$\bot$ or $\mathbf{false}$ or 0 for $(A_1 \wedge \neg A_1)$

Truth tables (II)

Tables for the operators $\rightarrow$, $\leftrightarrow$:

A	B	A	$\rightarrow$	B
0	0	0	1	0
0	1	0	1	1
1	0	1	0	0
1	1	1	1	1

Name: *implication*

Interpretation: If A holds, then B holds.

A	B	A	$\leftrightarrow$	B
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	1	1	1

Name: *equivalence*

Interpretation: A holds if and only if B holds.

Beware!!!

$A \to B$ does not say, that A is a cause of B.

"Penguins swim $\to$ Horses neigh"
is true (in our world).

$A \to B$ does not say anything about the truth value of A.

"Ms. Merkel is a criminal $\to$ Ms. Merkel should go to prison"
is true (in our world).

A false statement implies anything.

"Penguins fly $\to$ Mr. Obama is a criminal"
is true (in our world).

Formalizing natural language (I)

A device consists of two parts A and B, and a red light. We know that:

- A or B (or both) are broken.
- If A is broken, then B is broken.
- If B is broken and the red light is on, then A is not broken.
- The red light is on.

We use the atomic formulas: Ro (red light on), Ab (A is broken), Bb (B is broken), and formalize this situation by means of the formula

$$((((Ab \vee Bb) \wedge (Ab \rightarrow Bb)) \wedge ((Bb \wedge Ro) \rightarrow \neg Ab))) \wedge Ro)$$

Formalizing natural language (II)

Full truth table:

Ro	Ab	Bb	$((((Ab \vee Bb) \wedge (Ab \rightarrow Bb))) \wedge ((Bb \wedge Ro) \rightarrow \neg Ab)) \wedge Ro)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

Formalizing natural language (III)

Formalize the Sudoku problem:

4				9				2
		1				5		
	9		3	4	5		1	
		8				2	5	
7		5		3		4	6	1
	4	6				9		8
	6		1	5	9		8	
		9				6		
5				7				4

An atomic formula X_{YZ} for each triple $(X, Y, Z) \in \{1, \ldots, 9\}^3$:

$X_{YZ} =$ the square at row Y and column Z contains the number X

Example: The first row contains all digits from 1 to 9

$$\bigwedge_{X=1}^{9} \left(\bigvee_{Z=1}^{9} X_{1Z} \right)$$

The truth table has

$$\begin{aligned} 2^{729} = {} & 282401395870821749694910884220462786335135391185 \\ & 157752468340193086269383036119849990587392099522 \\ & 999697089786549828399657812329686587839094762655 \\ & 308848694610643079609148271612057263207249270352 \\ & 7723757359478834530365734912 \end{aligned}$$

rows.

Models

Let F be a formula and let $\mathcal{A}$ be an assignment. $\mathcal{A}$ is suitable for F if it is defined for every atomic formula A_i occurring in F.

Let $\mathcal{A}$ be suitable for F:

If $\mathcal{A}(F) = 1$	then we write	$\mathcal{A} \models F$
	and say	F holds under $\mathcal{A}$
	or	$\mathcal{A}$ is a model of F
If $\mathcal{A}(F) = 0$	then we write	$\mathcal{A} \not\models F$
	and say	F does not hold under $\mathcal{A}$
	or	$\mathcal{A}$ is not a model of F

Validity and satisfiability

Validity: A formula F is valid (or a tautology if *every* suitable assignment for F is a model of F. We write $\models F$ if F is valid, and $\not\models F$ otherwise.

Satisfiability: A formula F is satisfiable if it has at least one model, otherwise F is unsatisfiable.
A (finite or infinite!) set of formulas S is satisfiable if there is an assigment that is a model of every formula in S.

Exercise

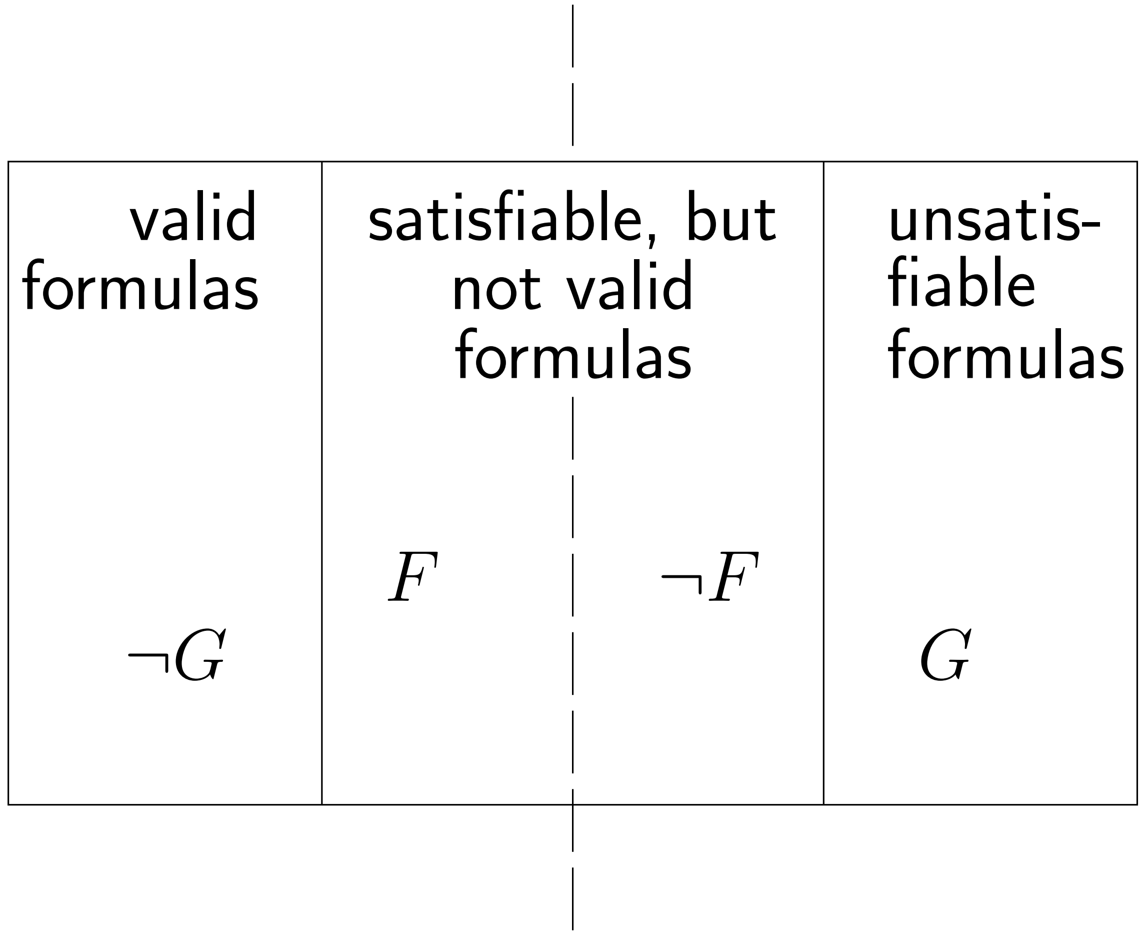
	Valid	Satisfiable	Unsatisfiable
A			
$A \vee B$			
$A \vee \neg A$			
$A \wedge \neg A$			
$A \rightarrow \neg A$			
$A \rightarrow B$			
$A \rightarrow (B \rightarrow A)$			
$A \rightarrow (A \rightarrow B)$			
$A \leftrightarrow \neg A$			

Exercise

Which of the following statements are true?

				Y/N	C.ex.
If	F is valid,	then	F is satisfiable		
If	F is satisfiable,	then	$\neg F$ is satisfiable		
If	F is valid,	then	$\neg F$ is satisfiable		
If	F is unsatisfiable,	dann	$\neg F$ is valid		

Mirroring principle



Consequence

A formula G is a consequence or follows from the formulas $F_1, \ldots, F_k$ if every model $\mathcal{A}$ of $F_1, \ldots, F_k$ that is suitable for G is also a model of G

If G is a consequence of $F_1, \ldots, F_k$ then we write $F_1, \ldots, F_k \models G$.

Consequence: example

$$(Ab \vee Bb), (Ab \to Bb),$$
$$((Bb \wedge Ro) \to \neg Ab), Ro \quad \models \quad ((Ro \wedge \neg Ab) \wedge Bb)$$

Exercise

M	F	$M \models F$?
A	$A \vee B$	
A	$A \wedge B$	
A, B	$A \vee B$	
A, B	$A \wedge B$	
$A \wedge B$	A	
$A \vee B$	A	
$A, A \rightarrow B$	B	

Consequence, validity, satisfiability

The following assertions are equivalent:

1. $F_1, \ldots, F_k \models G$, e.g. , G is a consequence of $F_1, \ldots, F_k$.
2. $((\bigwedge_{i=1}^{k} F_i) \rightarrow G)$ is valid.
3. $((\bigwedge_{i=1}^{k} F_i) \wedge \neg G)$ is unsatisfiable.

Exercise

Let S be a set of formulas, and let F and G be formulas. Which of the following assertions hold?

	Y/N
If F satisfiable then $S \models F$.	
If F valid then $S \models F$.	
If $F \in S$ then $S \models F$.	
If $S \models F$ then $S \cup \{G\} \models F$.	
$S \models F$ and $S \models \neg F$ cannot hold simultaneously.	
If $S \models G \to F$ and $S \models G$ then $S \models F$.	